# 附网存储设备用户行为的一种层次化免疫策略*

孙照焱，董永贵，贾惠波，冯冠平
(清华大学 精密仪器与机械学系 精密测试技术及仪器国家重点实验室，北京 100084)

**摘　要**: 受生物免疫机制的启发,针对附网存储设备用户的异常行为,提出由用户认证、文件权限和用户阶梯组成的层次化免疫策略,对用户行为所请求的系统调用序列进行异常监控,并实现了基于系统调用对和用户分阶信息的异常检测算法,特征元素的阶梯式矩阵存储和高效的匹配方法保证了免疫策略的快速实施。实验表明,该策略能够有效阻止非法用户的入侵及合法用户的越权访问,且足够快的响应速度完全满足在线检测的需要。

**关键词**: 附网存储设备; 入侵检测系统; 系统调用; Linux
**中图法分类号**: TP309　　**文献标识码**: A　　**文章编号**: 1001-3695(2005)01-0111-03

# A Multilevel Immune Strategy for User Behaviors in Network-Attached Storage Device

SUN Zhao-yan, DONG Yong-gui, JIA Hui-bo, FENG Guan-ping
*(State Key Laboratory of Precision Measurement Technology & Instruments, Dept. of Precision Instruments & Mechanology, Tsinghua University, Beijing 100084, China)*

**Abstract:** Inspired from the biological immunity mechanism, a multilevel immune strategy, composed of user authentication, access authority of file system and user stair, is presented to identify abnormal behaviors in network-attached storage devices. Tracking the system calls required bye users' operations, the anomalies are monitored. An anomaly detection algorithm, which is based on system call pairs and user rank, is established and implemented. The eigenvalues are stored in a novel matrix and an efficient matching method is utilized, which ensures the immune strategy to be carried out rapidly. Experimental results show that this strategy can abort anomalies efficiently, including intrusions of unauthorized users and inadmissible accesses of authorized users. Furthermore, the response speed is fast enough for on-line monitoring.

**Key words:** Network-Attached Storage Device(NASD); Intrusion Detection System(IDS); System Call; Linux

## 1　引言

随着数字化信息的剧增以及网络的飞速发展,数据的存储逐渐从单机转向网络。附网存储设备(Network-Attached Storage Device, NASD)利用优化的 Linux 操作系统,将物理存储介质(如磁盘、光盘等)直接与快速以太网相连,为客户机提供文件共享服务。NASD 在局域网中的应用为用户共享资源提供了便利,但是由于其体系结构和访问协议等原因,数据的访问安全成了薄弱环节。入侵检测是计算机系统在访问控制、身份识别和认证、密码技术及防火墙之后的又一道安全保护屏障,它是主动实现系统安全防护的一种重要方法。目前常见的入侵检测技术需要处理大量的信息,计算烦琐、实时性差,不适于应用在 CPU 资源主要用于处理文件请求的 NASD 中。近年来,由生物免疫系统启发的人工免疫系统(Artificial Immune System, AIS)引起了国际上许多学者的极大兴趣[1],该领域的一个主要研究方向就是计算机网络的入侵检测及安全防护[2]。生物免疫系统是高度进化的智能系统,它具有分布式、自适应以及动态平衡等特点,具备学习、记忆和识别功能。AIS 借鉴这些特点及功能,可以建立人工免疫网络模型[3],开发免疫学习算法等。Forrest 与 Anil Somayaji[4] 把 AIS 引入 Linux,建立进程动态平衡(pH)系统,针对系统内部的系统调用序列进行检测,以实现入侵检测。这种针对通用 Linux 系统的 pH,考虑的是广泛意义上的非法用户入侵,忽略了系统合法用户之间访问权限的差异,因此对于 NASD 并不适用。

NASD 的用户行为比较简单,主要是通过局域网对文件的读取和存储,不同类型的用户拥有不同级别的文件访问权限。参考生物分层的免疫体系结构[5],本文提出一种层次化的免疫策略,包括用户认证、文件权限和用户阶梯三个部分。该策略将不同类型用户的行为限制在不同的阶梯中,采用一种快速有效的监控算法,实时地检测并禁止系统内部异常的功能调用,防止非法用户的入侵以及合法用户的越权行为。

## 2　分层免疫策略

生物免疫系统识别机体中的"本体(Self)"和"异体(Non-self)",并排除"异体",以维持生物体的生理平衡[6],这里将用户合法的正常行为定义为"本体",而非法的异常行为定义为"异体"。通常 Linux 系统对用户的权限管理依赖于两道屏障:用户认证和文件权限。用户认证需要用户的账号和密码,而文件系统的许可机制则从文件和目录两级给不同用户设置了不同的访问权限(读、写、执行),这两层防护相当于生物体的皮肤和固有免疫两层免疫机制。尽管这样,一些别有用心的入侵

收稿日期: 2004-02-29; 修返日期: 2004-04-25
基金项目: 国家重点基础研究"973"基金资助项目(G19990330)

者仍然可能破译用户的密码,利用系统的某些漏洞进行非法操作。在 NASD 系统中,用户的操作命令产生一个或多个进程,而进程又根据功能需要进一步请求系统最底层的系统调用,最终用户的命令是由系统调用组成的序列来完成的。Linux 内核定义的系统调用是系统运行的基本单位,数量有限,约两百多个,而 NASD 用户存取文件的操作所涉及到的系统调用数量就更少了。入侵者的异常行为"异体"会产生与合法用户正常行为"本体"不同的系统调用请求,因此监控用户操作所请求的系统调用,识别"异体"并对其进行免疫是一种行之有效的策略。

在用户认证和文件权限基础上,分层免疫策略设置了第三道屏障——用户阶梯,实施对系统异常操作的获得免疫,如图 1 所示。穿过前两道屏障的用户操作附带着用户的必要信息,如用户号、组号等,用户分阶据此把用户分为普通用户、管理员和系统用户等。不同的用户处于不同的阶梯上,在相应的阶梯中保存有该阶用户正常的、合法的行为模式库(即"本体"库)。用户的操作命令按照分阶被发送到各自对应的阶梯中,分布于各阶梯中的入侵检测系统(Intrusion Detection System, IDS)则分析用户命令请求的系统调用,识别"本体"和"异体"。若判断用户行为属于"本体",则让其穿过第三道屏障被系统执行;若判断为"异体",则进行免疫,禁止运行。IDS 包括数据提取、数据存储、入侵分析和响应处理四个部分,如图 2 所示。

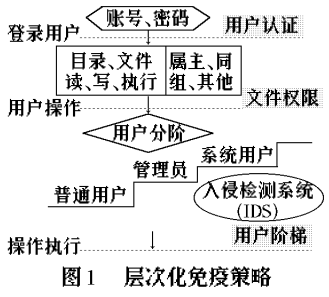

图 1 层次化免疫策略

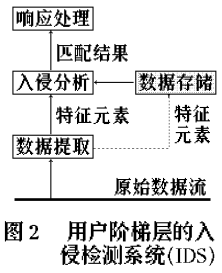

图 2 用户阶梯层的入侵检测系统(IDS)

数据提取模块是系统的数据采集器,负责提取反映系统运行状态的数据并进行预处理。虽然 NASD 系统中的系统调用总共只有两百多个,但是各个进程所请求的系统调用的数量、次序各不相同,因此一个操作命令产生的系统调用序列可能非常长(在试验过程中,一个通过网络上载文件的操作请求了 4 000 多次系统调用)。如果收集完进程所请求的所有系统调用之后再进行"本体"、"异体"的分析,那样就无法做到实时免疫。采用 Anil Somayaji[7] 提出的沿时间轴加窗滑动的方法,对系统调用序列加窗,提取窗内的系统调用对作为检测的依据。由于用户阶梯对用户类型进行了区分,因此把系统调用对附加上用户的分阶信息,才能作为入侵分析的特征元素。

数据存储模块保存各阶梯用户的合法行为信息库,也就是"本体"的集合。库中的特征元素是数据提取模块在先期的数据收集阶段采集到的,存储在一个 256 ×256 规模的矩阵中。

入侵分析模块用来识别"本体"和"异体",从数据提取模块接收当前系统运行所产生的特征元素,将其与行为库中对应阶梯内的"本体"数据进行匹配,将分析结果发送给响应处理模块。入侵分析的实时性直接受到数据存储方式以及匹配算法的影响,优化的数据存储方式可以大大缩减特征元素查找匹配的时间,第三部分将详细介绍阶梯式的矩阵存储方式和快速匹配算法。

响应处理模块则根据分析结果("本体"或者"异体")来决定该操作是否执行。对于"异体",采取主动响应措施进行免疫,禁止其运行,同时记录相关的信息供管理员查询,如用户号、操作命令等。

## 3 入侵检测算法

### 3.1 数据提取

在建立"本体"模式库和进行入侵分析时,都需要从系统调用序列中提取特征元素。"本体"库的建立可以像生物免疫系统那样从已有的库复制克隆而来,也可以在一个安全的环境中,进行一段时间的训练,遍历各种操作,从中提取出特征元素,存储在模式库中。为了便于说明如何进行数据提取,先给出下列约定:

- 滑动窗长 $w$——滑动窗内所包含的系统调用个数。
- 当前调用 $c$——窗最右边的一个系统调用,系统调用按被请求的时间先后顺序从左到右排列。
- 系统调用对 $p$——窗内除 $c$ 之外的每一个系统调用和 $c$ 的组合,个数为 $(w-1)$。
- 调用对距离 $l$——调用对 $p$ 中两个系统调用在窗内位置的距离,最小为 1,最大为 $(w-1)$。
- 用户阶数 $n$——各阶用户在阶梯上所处的阶数,图 1 中普通用户、管理员分别位于第 0,1 阶上。
- 用户阶高 $h$——用户分阶信息 $n(w-1)$,对应于各阶用户的特征信息在模式库中的起始位置。

"open, fstat, mmap, read, mmap, read, read, close, munmap, uname, stat, …"是一个普通用户($n=0$)通过网络进行文件存取过程中记录的一段系统调用序列。系统内核为每一个调用分配了一个独立的编号,在内核 2.2.19 中从 1 分配到 225,这段序列对应的编号为"5, 108, 90, 3, 90, 3, 3, 6, 91, 122, 106, …"。选取窗长 $w$ 为 9,沿着系统调用序列从左向右滑动,得到加窗后的系统调用序列,如表 1 所示。表中每一行是一次加窗的结果,窗最右边的对象是当前的系统调用 $c$,最左边的对象是 $c$ 之前的第八个系统调用,其余类推,几个当前调用相同的窗进行了合并。表 1 中,如果"open"是被请求的第一个系统调用,则它前面空出的位置将补上编号"255"(定义 255 号系统调用为空调用)。需要提取的特征元素信息为 $[p(j,i), l, h]$,表 1 中灰色底纹行所表示的窗的编号表示为(255, 5, 108, 90, 3, 90, 3, 3, 6),当前调用 $c=6$,则从中提取出的八个特征元素如表 2 所示。

表 1 加窗后的系统调用序列

| 前调用 8 | 前调用 7 | 前调用 6 | 前调用 5 | 前调用 4 | 前调用 3 | 前调用 2 | 前调用 1 | 当前调用 c |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Open |
| | | | | | | | Open | Fstat |
| | | | | Open | Fstat | Open,Mmap | Fstat,Read | Mmap |
| | | Open | Open,Fstat | Fstat,Mmap | Open,Mmap,Read | Fstat,Read,Mmap | Mmap,Read | Read |
| | Open | Fstat | Mmap | Read | Mmap | Read | Read | Close |
| Open | Fstat | Mmap | Read | Mmap | Read | Read | Close | Munmap |
| Fstat | Mmap | Read | Mmap | Read | Read | Close | Munmap | UnMap |
| Mmap | Read | Mmap | Read | Read | Close | Munmap | Uname | Stat |

表 2 加窗提取的特征元素

| $p(j,i)$ | $l$ | $h$ |
|---|---|---|
| 255* * * * * * * 6 | 8 | 0 |
| 5* * * * * * 6 | 7 | 0 |
| 108* * * * * 6 | 6 | 0 |
| 90* * * * 6 | 5 | 0 |
| 3* * * 6 | 4 | 0 |
| 90* * 6 | 3 | 0 |
| 3* 6 | 2 | 0 |
| 3 6 | 1 | 0 |

### 3.2 数据存储

对提取出的特征元素,如果采用常规的多维数组方式保

存, 则一个特征元素需要占用四个字节。而且在进行“本体”、“异体”的识别时, 将花费不少时间从数组中查询合法行为的特征元素信息。根据 NASD 系统中系统调用编号连续的特点, 采用阶梯式的矩阵存储方式来保存数据, 可以只用字节中的一位来保存一个特征元素的信息。

定义一个 256 ×256 规模的二维数组 $E$, 该二维数组相当于一个存储矩阵, 其行号和列号都对应于系统调用的编号(0 ～255, 存在部分冗余编号), 矩阵的每个位置上的元素保存了其行列号对应的系统调用组成的 $p$ 的信息, 初始化时全为 0。假定系统有四类用户, 即用户阶数 $n$ 分别为 0, 1, 2, 3, 窗长 $w=9$, 则定义 $E$ 中每一个元素为四个字节(32 位)长度。用户阶高 $h$ 分别为 0, 8, 16, 24, 每一级阶梯内有一个字节(八位)的空间保存该阶用户的合法调用信息, 不同的位分别对应不同的距离 $l$。对于一个阶高为 $h$、距离为 $l$、由系统调用 $i$ 和 $j$ 组成的系统调用对 $p$, 它的存储结果是 $E[i][j]$ 的 bit$(l-1+h)=1$ (图 3)。

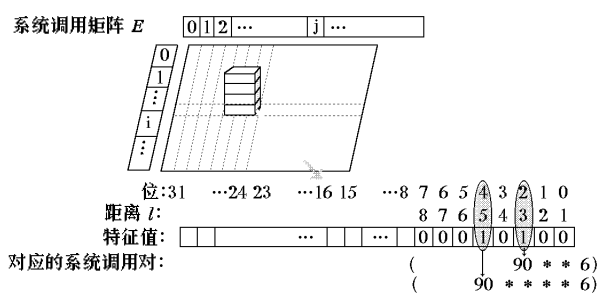


图 3　特征元素的阶梯式存储

把表 2 中距离 $l$ 为 3, 5 的两个特征元素[(90, 6), 3, 0], [(90, 6), 5, 0]保存在 $E$ 中, 结果如图 3 所示。系统的当前调用 $c=i=6$, 其前调用 3 和前调用 5 的编号均为 $j=90$, 用户阶高 $h=0$, 距离 $l$ 分别为 3 和 5, 则元素 $E[6][90]$ 的 $bit2=bit4=1$。对于 0 阶用户, $E[i][j]=E[6][90]=00010100$ (低八位)。同样, 表 2 中编号为 $j=3$ 的系统调用和 $c$ 组成的 $p$ 存储结果为: $E[i][j]=E[6][3]=00001011$(低八位)。

### 3.3　匹配算法

由于各阶用户的合法行为都以阶梯的方式保存在矩阵 $E$ 中, 因此对当前系统进行入侵检测时, 入侵分析模块根据提取的特征元素 $[p(j, i), l, h]$, 到矩阵中查看元素 $E[i][j]$ 对应位 bit $(l-1+h)$ 的值是否为 1, 若为 1, 说明待检测的系统调用对是合法的, 否则是非法的。这种快速的匹配算法大大节省了模式库特征值查找的时间开销, 使系统得以实时地运行。对于当前的系统调用 $c$, 滑动窗内 $(w-1)$ 个系统调用对都是合法的, 才可以判定 $c$ 合法, 即为“本体”; 如果有一个系统调用对非法, 就可以判定 $c$ 为非法调用, 即为“异体”。之所以采用如此严格的分析规则, 是因为 NASD 在局域网中通常扮演功能专一服务器的角色, 各个用户阶梯中用户的合法行为模式是比较固定的。“本体”模式库建立以后, 基本不需要更新, 如果有新的系统调用出现, 则可以认为不是系统应该执行的操作。

### 3.4　实验结果

为了验证 IDS 的检测效果, 在 NASD 系统中设置了一个普通用户的账号“nastest”, 用户号为 501。然后仅仅将该用户读取文件的操作提取为“本体”保存在模式库中, 于是当 nastest 登录系统, 企图使用“rm”命令删除一个文件时, 该操作被识别为“异体”, 禁止运行。从系统的消息文件(/var/log/messages)中可以看到 IDS 记录下来的非法操作信息:

…

Nov 26 20:25:20 tu053139 kernel: current task is stopped.

Nov 26 20:25:20 tu053139 kernel: syscall = 122, uid = 501, gid = 501, command = rm

…

分层的用户行为免疫策略对用户操作所请求的每一个系统调用都进行检测, 无疑会增加系统执行命令的时间。由于采用了上述快速匹配算法, 系统入侵的实时检测得到了保证。表 3 列出了采用免疫检测方法的系统和普通系统在文件传输性能方面的比较, 包括 CPU 占有率、内存使用和传输时间三项内容(测试系统的配置为: PⅢ 1GHz, 256MB SDRAM)。测试结果表明, 在局域网内的 NASD 中加入用户阶梯层, 对系统调用免疫检测, 虽然增加了少量的 CPU 占用率(总 CPU 占有率仍然较低), 但是在文件传输时间方面并没有大幅度的增长。读取多个小文件花费时间所受影响最大, 但也仅仅增加了 1.81%, 少于 pH 对系统性能的 5% 降低[7], 这是因为 pH 考虑的是通用 Linux 系统中用户的异常行为, 用户多, 行为复杂, pH 需要监测众多用户的各个应用程序。而 NASD 系统功能单一, 用户行为简单, 系统只需要针对文件请求是否合法来进行检测, 因此这种异常检测方法具有更好的实时性。

表 3　免疫检测的实时性测试

| | 单个大文件 (一个文件 730.78MB) | | | | | | 多个小文件 (2 000 个文件共 281.66MB) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 读 | | | 写 | | | 读 | | | 写 | | |
| | CPU (%) | 内存 (kB) | 时间 (s) | CPU (%) | 内存 (kB) | 时间 (s) | CPU (%) | 内存 (kB) | 时间 (s) | CPU (%) | 内存 (kB) | 时间 (s) |
| 普通系统 | 18.4 | 254356 | 94.44 | 19.4 | 254444 | 91.13 | 12 | 254596 | 269.69 | 15.6 | 254604 | 106.66 |
| 免疫系统 | 20.8 | 254656 | 94.63 | 21.2 | 254664 | 92.13 | 12.2 | 254588 | 274.56 | 17.6 | 254216 | 107.25 |
| 增长(%) | 13.04 | 0.12 | 0.20 | 9.28 | 0.09 | 1.10 | 1.67 | 0 | 1.81 | 12.82 | 0 | 0.55 |

## 4　结束语

受生物免疫系统分层体系结构和对“本体”、“异体”识别机制的启发, 提出一种针对用户异常行为的多层免疫策略, 对局域网中 NASD 系统的用户访问行为进行管理。用户的账号和密码形成第一道保护屏障, 而文件访问权限则是系统的第二道安全防线。用户阶梯作为第三道防线, 在用户操作命令运行的最底层对系统调用进行监控。部署于各个用户阶梯中的 IDS 根据“本体”模式库, 判断用户请求的系统调用是否合法, 以此识别“本体”和“异体”。对于“异体”行为, 免疫应答机制禁止其运行以确保系统不被侵害。由于用户阶梯限制了各阶用户只能在“本体”库授权范围内进行操作, 即使用户利用系统漏洞获取了其他阶用户的访问权限, 也无法进行越权访问。

相对于通用服务器来说, 应用于局域网中的 NASD 功能专一, 用户行为比较简单, 具有固定的访问模式。特征元素巧妙的存储方式和快速的匹配算法使 IDS 具有良好的实时性。实验表明, 这种免疫策略能够有效地阻止非法用户的入侵以及合法用户的越权行为, 而且实时性强, 对 NASD 文件传输性能的降低不足 2%。这种方法在网络存储领域中的应用, 可望解决资源共享和信息访问安全之间的矛盾。

(下转第 116 页)